

2.1 Part 1: Viterbi Decoding

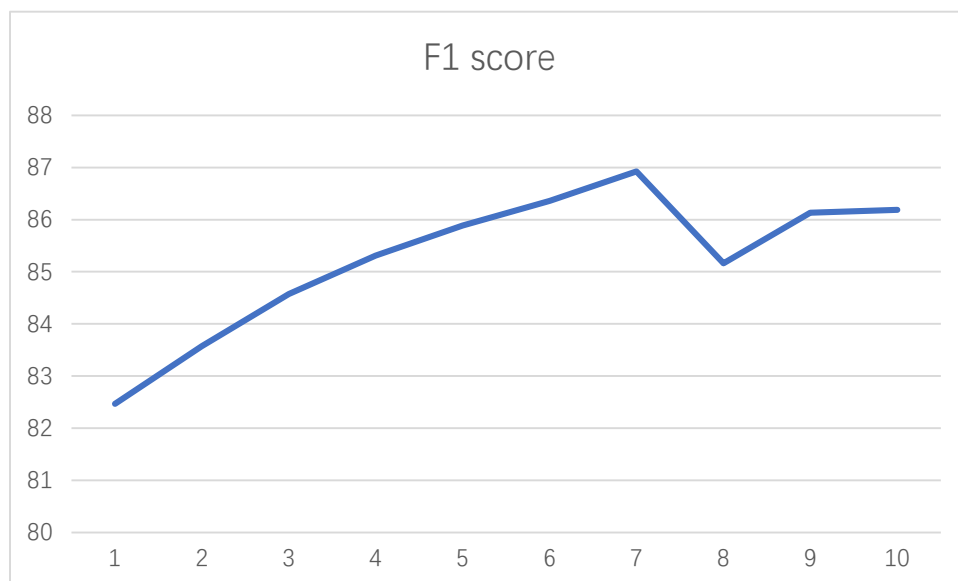
Results:

```
Labeled F1: 75.72, precision: 4218/5198 = 81.15, recall: 4218/5943 = 70.97
```

2.2 Part 2: CRF Training

1. For the emission features, I use the given emission features.
2. For the transition weights, I modify the weights from the HMM model to meet the hard constraints that only B-X or I-X can transition to I-X. For a “to” entry I-X, any weight whose “from” entry is not B-X or I-X is set to -1000. The weights are attached as “transition.csv”.
3. I implemented Adam as my optimizer.
4. I trained the CRF model for 10 epochs using 2.14 hours, and attach my training log “NER_train.o5369960”.

After every training epoch, I save the model parameters and test it on the development set. Here is the F1 score chart:



You can see that after the 7th epoch, my model achieves the best performance on the development set, as shown below:

```
Labeled F1: 86.93, precision: 5130/5859 = 87.56, recall: 5130/5943 = 86.32
```

Thus, I decide to use the model the 7th epoch as the final CRF model. The parameters are attached as “model_7.npy”. The CRF’s output on the testb blind test set in CoNLL format is attached.